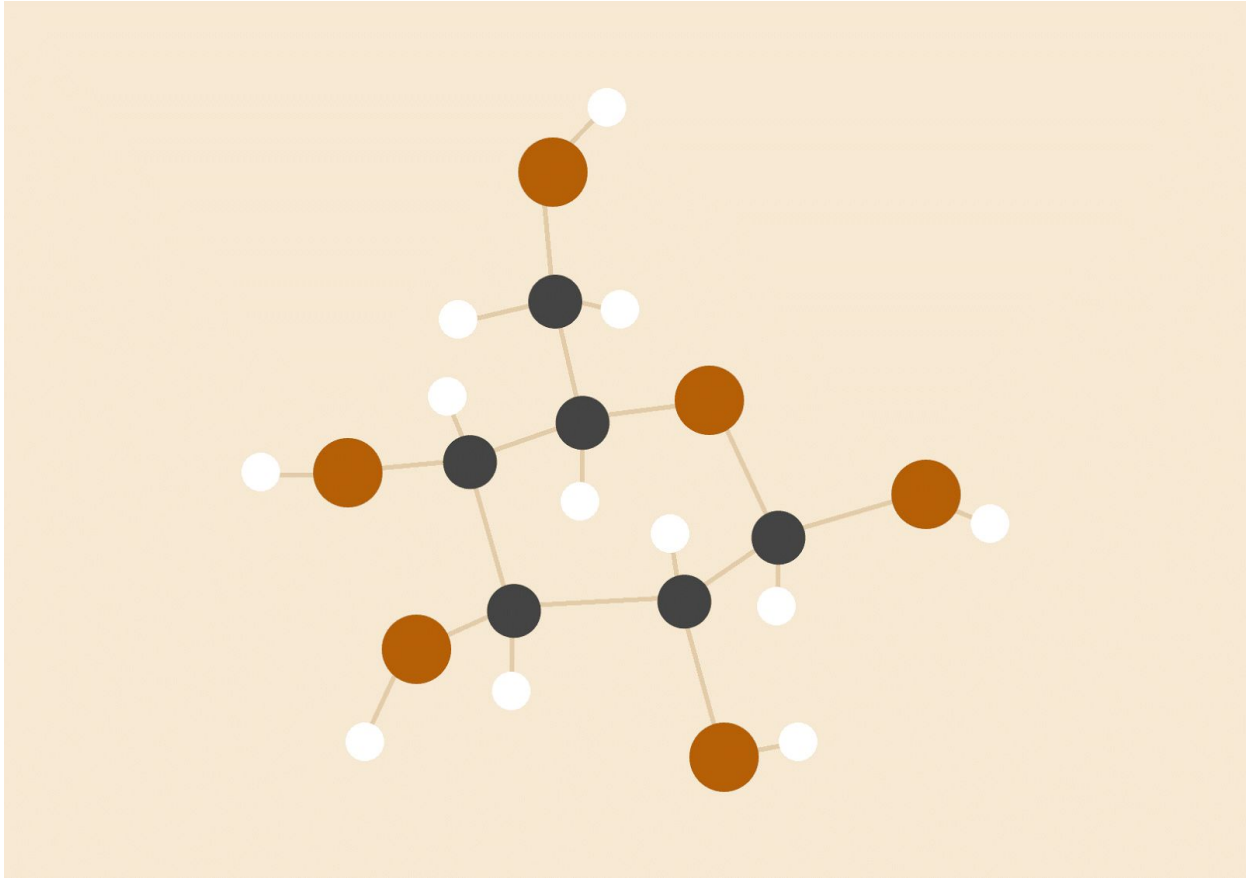


Red Hat Subscription and Entitlement Accounting

with generic examples



Red Hat

Dan Lah

5/26/2016

Version 1.1

Introduction

Subscriptions unlock valuable assets, but providing flexibility in an ever expanding product base and deployment possibilities quickly becomes a challenge, especially when old and new subscriptions are mixed. This article details the accounting of Red Hat subscriptions.

SKU vs. Products and Subscription vs. Entitlement

As a customer of Red Hat you want access to updates and knowledge for Red Hat **products**, like Red Hat Enterprise Linux. The access to those products is provided by a **subscription**. To purchase a subscription you procure a **SKU** (stock keeping unit) which ties the subscription to an “entitlement” quantity and price. The subscription entitles your account to Red Hat services and material content through the Customer Portal (see the value of a subscription here: <https://access.redhat.com/subscription-value>).

The subscription also provides end-systems access to update content from Red Hat’s datastore. Since a single subscription may provide content to multiple end-systems, the subscription is converted into at least one pool of **entitlements**. Ultimately, entitlements are attached to an end-system which results in entitlement certificates being granted to the end-systems. The right to access updated Red Hat product content requires entitlements.

Types Of Subscriptions

Due to evolving subscription designs in response to changing business needs and a desire to reduce the total number of subscriptions that Procurement must deal with, Red Hat subscriptions can be one of several types. Two predominate types are “**standard**” and “**instance-based**”.

- ❖ The original “standard” subscription entitled just one end-system of a particular deployment (physical vs. virtual). But the customer had to manage multiple subscriptions to enable the same product in different deployments.
- ❖ As Red Hat’s product base grew there became scenarios where it was better for a single “standard” subscription to have a quantity greater than one and where deployment of the end-system (physical vs. virtual) did not matter. The end result was instead of a customer having to purchase and manage multiple subscriptions a single subscription could entitle multiple end-systems.
- ❖ Ultimately, scenarios arose where the ability to be flexible in the deployment of the end-system (physical vs. virtual) was desired, but the deployment **did** matter. The end result was the introduction of the “instance-based” subscription that is flexible to be used on a physical or a virtual system, but the quantity supported is dependent on the deployment.

Converting Subscriptions into Pools of Entitlements

In implementation, the subscription service (Customer Portal for hosted connectivity or Satellite 6 for on-premise connectivity) converts each subscription into a “pool” of individual entitlements that can be attached to an end-system. The attachment process “consumes” an entitlement from the pool thereby “entitling” the end-system to access content stores. When creating the pools the total number of entitlements in the pool is affected by:

- ❖ **subscription quantity** - number of subscriptions purchased
- ❖ **entitlement quantity** - number embedded in the subscription with indicates how many end-systems the subscription the can entitle (usually in the subscription’s title)
- ❖ **subscription type** - an embedded dependency on the end-system’s deployment and includes an “*instance multiplier*”:
 - If the subscription is of “standard” type: the number of entitlements in the pool equals the subscription quantity multiplied by entitlement quantity.
 - If the subscription is of “instance-based” type: the number of entitlements in the pool equals the subscription quantity multiplied by the entitlement quantity multiplied by the *instance multiplier* (which represents the number of virtual guest systems that could alternatively be entitled in place of one physical system) .

In older “standard” subscriptions the entitlements quantity ended up the same as the subscriptions quantity. But with “standard” subscriptions extended to support multiple end-systems and the “instance-based” subscriptions supporting multiple end-systems dependent on deployment, the calculated conversion of subscriptions to entitlements is now a central part of understanding Red Hat subscriptions.

Subscription	Subscription Type	Subscription Quantity Purchased		Entitlement Quantity in Subscription		Instance Multiplier in Subscription		Total Entitlements In Pool
RH0103678* , Red Hat Enterprise Linux Server, Premium (1-2 sockets) (Up to 1 guest) with Smart Management	Standard	1	*	1	*	1	=	1
RS00013 , Red Hat Ceph Storage for Management Nodes, Premium (Up to 6 Physical Nodes)	Standard	1	*	6	*	1	=	6
RH00008** , Red Hat Enterprise Linux Server with Smart Management, Premium (Physical or Virtual Nodes)	Instance Based	1	*	1	*	2	=	2

* For brevity, this example does not detail this subscription’s offering of a “bonus guest”, but acknowledges that the bonus guest follows the same strategy as a “standard” subscription but uses a different attribute from the subscription to calculate the entitlements. See below for information on bonus guests.

** The RH00008 subscription is similar to the RH00003 subscription; however RH00008 offers smart management.

Consumption Rules (Rates)

Entitlements are “attached” to end-systems to entitle the system to Red Hat content. The attachment of the entitlement “consumes” entitlement from the pool thus reducing the number of entitlements available to entitle other end-systems.

Each subscription has a consumption rule that determines the number of entitlements (rate) that are consumed from a pool to cover the end-system. The consumption rule depends on several factors including: the subscriptions type, the “**unit of capacity**” the subscription covers (e.g. sockets, ram, cores), and the deployed system type (physical or virtual); thus the consumption rate is governed by the following statements:

- ❖ **“Standard” subscriptions consumed entitlements at a rate of:**
 - **One per each virtual system deployment**
 - **One per unit of capacity on the physical system**
- ❖ **“Instance-based” subscriptions consumed entitlements at a rate of:**
 - **One per each virtual system deployment**
 - **The “instance multiplier” per unit of capacity on the physical system**

For now we will assume that each physical end-system is a two-socket system, so from these consumption rules the end-systems support matches the number of entitlements in the pools.

Subscription	Subscription Quantity Purchased	Number of End-Systems Supported
RH0103678* , Red Hat Enterprise Linux Server, Premium (1-2 sockets) (Up to 1 guest) with Smart Management*	1	1 Physical System
RS00013 , Red Hat Ceph Storage for Management Nodes, Premium (Up to 6 Physical Nodes)	1	6 Physical Systems
RH00008** , Red Hat Enterprise Linux Server with Smart Management, Premium (Physical or Virtual Nodes)	1	2 Virtual Systems -OR- 1 Physical System

* For brevity, this example does not detail this subscription’s offering of a “bonus guest”, but acknowledges that the bonus guest follows the same strategy as a “standard” subscription but uses a different attribute from the subscription to calculate the entitlements. See below for information on bonus guests.

** The RH00008 subscription is similar to the RH00003 subscription; however RH00008 offers smart management.

Multiple Pools

Subscriptions are always initially managed through the Customer Portal. The customer can manage the entitlements of the subscriptions either with the Customer Portal or with Satellite. Regardless of where the entitlements are managed, entitling the end-systems follow the same workflow:

- ❖ The end-system registers with a subscription service
- ❖ The end-system then attaches a subscription (entitlement)

In the example below the displays are from the subscription service client known as subscription-manager using the command to “list all”. The end-system is a physical system and is registered to the Customer Portal.

Here a end-system is offered **two pools** created from two different subscriptions above. Two pools are offered since both can be used to cover the product and capacity of the system. The subscription on the left is a “standard” and the one on the right is an “instance-based”. Note that subscription-manager shows not only the “**available**” entitlements in the pool, but recognizes the type and consumption rate of the subscription in the pool and thus “**suggests**” a number to attach to cover the end-system.

Had the end system been a virtual system the display on the left would be unchanged, however, the display on the right would change to “Suggested: 1” since the consumption rate is different in “instance-based” subscriptions between physical and virtual end-systems.

Subscription Name: Red Hat Enterprise Linux Server, Premium (1-2 sockets) (Up to 1 guest) with Smart Management		Red Hat Enterprise Linux Server, Premium (Physical or Virtual Nodes)	
SKU:	RH0103678	SKU:	RH00008
Contract:	xxxxxxx	Contract:	xxxxxxx
Pool ID:	xx	Pool ID:	xx
Management:	Yes	Management:	Yes
Available:	1	Available:	2
Suggested:	1	Suggested:	2
Service Level:	Premium	Service Level:	Premium
Service Type:	L1-L3	Service Type:	L1-L3
Subscription Type:	Standard*	Subscription Type:	Instance Based
Ends:	04/10/2017	Ends:	04/10/2017
System Type:	Physical	System Type:	Physical

For brevity, this example does not detail the subscription RS00013S, however, the behavior would be similar to the “standard” offering, but would indicate there are 6 entitlements available.

SMA and Manifests

When using Satellite, before the end-systems can register and attach entitlements, the Satellite must charge-up its entitlement pools. This is done through a Subscription Management Application (SMA) in the Customer Portal and selecting entitlements from the Customer Account into the SMA. This creates a manifest which is used to establish the entitlement pools in the on-premise Satellite.

What is being selected into the SMA are entitlements from the entitlement pools in the Customer Portal, not the subscriptions, thus it is important to recognize the type of subscription and the amount of entitlements that are being selected since moving a quantity of one of a “standard” subscription sufficiently may move enough entitlements to cover a physical system, but moving a quantity of one of an “instance-based” subscription will only provide enough entitlements to cover a virtual system.

For the example below, in the Customer Portal a SMA was created by moving a quantity of “1” from each subscription into the SMA and then using the corresponding manifest to create the pools in a Satellite. If the same physical end-system from the above example registered to the Satellite instead of the Customer Portal, the results are different. The resulting pool for the “instance-based” and “standard” subscriptions only have one entitlement. The “standard” subscription can cover a virtual and/or a physical system, but the “instance-based” subscription pool only has enough entitlements to cover a virtual system (to cover the physical system “2” entitlements would have had to be added to the SMA before the manifest was exported).

Subscription Name: Red Hat Enterprise Linux Server, Premium (1-2 sockets) (Up to 1 guest) with Smart Management		Red Hat Enterprise Linux Server, Premium (Physical or Virtual Nodes)	
SKU:	RH0103678	SKU:	RH00008
Contract:	xxxxxxx	Contract:	xxxxxxx
Pool ID:	xx	Pool ID:	xx
Management:	Yes	Management:	Yes
Available:	1	Available:	1
Suggested:	1	Suggested:	2
Service Level:	Premium	Service Level:	Premium
Service Type:	L1-L3	Service Type:	L1-L3
Subscription Type:	Standard*	Subscription Type:	Instance Based
Ends:	04/10/2017	Ends:	04/10/2017
System Type:	Physical	System Type:	Physical

For brevity, this example does not detail the subscription RS00013S, however, the behavior would be similar to the “standard” offering, but would indicate there are 6 entitlements available.

Unit of Capacity and Stacking

Each subscription has a “**unit of capacity**” associated with it that describes a measurable attribute on the end-system that the subscription covers, such as, sockets or cores. RHEL Platform subscriptions have standardized on a socket-pair (two sockets) for physical systems or a single instance for virtual system. JBoss Middleware subscriptions have standardized on a “core” for both physical and virtual systems.

Thus when attaching a subscription to a system, the subscription service will evaluate the subscription’s capacity coverage against the system’s capacity and deployment, and set a status for the system. A system without a subscription will result in a status of “red”. If the system’s capacity is covered completely by the entitlement then the system’s status will be “green”. If there is a deficiency in the capacity provided by the attached subscription(s) needed to satisfy the capacity of the end-system, then the resulting system status will be yellow.

The most common status issue is under-entitling a system. This occurs when the entitlement coverage does not completely cover the system’s capacity; for example, a subscription that covers only a single socket pair is attached to a system with four sockets.

In older subscription models, Red Hat offered different subscriptions for specific capacities, such as, 2, 4, or 8 socket subscriptions. This not only lead to a proliferation of subscriptions that Procurement had to manage it also limited Operations, since a 4-socket subscription was intended for a 4-socket system. If the 4-socket subscription was attached to a 2-socket system, then the system would be over-entitled. Likewise the system would be under-entitled if the system had more than 4-sockets.

In an effort to reduce the number of subscriptions and make subscription consumption more flexible, newer subscriptions have the “**stacking**” attribute. This allows multiple entitlements from the same subscription (or compatible subscriptions) to be stacked onto a system to efficiently cover the capacity of the system.

The subscription service will recognize stacking when a subscription offers it and will suggest an efficient quantity to be attached. This is most noticeable in the list of available pools from the subscription-manager command line tool. Here you will see a “suggested” field showing a quantity that will be attached.

Shown below is a consumed subscription for a physical 4-socket system after attaching subscription RH00008. Here the output after using auto-attach indicates that a quantity of 4 was used. RH00008 is an instance-based subscription that will supply 2 entitlements to cover 2-sockets (a socket-pair) worth of capacity on the system. Since this system is a 4-socket system the subscription service “stacks” on an additional 2 entitlements to cover the system completely.

```
4-socket, physical system coverage by auto-attach using RH00008
subscription-manager list --consumed
Consumed Subscriptions
Subscription Name: Red Hat Enterprise Linux Server with Smart Management, Premium (Physical or Virtual
Nodes)
Provides: Red Hat Beta
SKU: RH00008
Contract: XXXXXXXX
Account: XXXXXXXX
Serial: XXXXXXXXXXXXX
Pool ID: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Provides Management: Yes
Active: True
Quantity Used: 4
```

Bonus Guests

Some RHEL subscriptions, carry an attribute called “*virt-limit*” which offers additional guest entitlements (sometimes called “bonus guests”) that are separate from the physical entitlement provided in the subscription. The physical entitlement is used for the hypervisor and the bonus guest entitlements are available for a specific number of virtual guests on that host.

In subscriptions prior to 2013, subscriptions may have offered bonus guests in various quantities such as 2, 4, 8, 16, and unlimited. However, cluster support with guest migration was very difficult when the nodes in the cluster all had different bonus guest support. The contractual obligations were difficult to understand and enforce by the customer.

The bonus guest strategy was changed in the 2013 subscriptions to simplified the offerings. Only unlimited guests were offered with the RHEL for Virtual Datacenters (vDC) subscription (RH00001); for deployments of high guest density. For other deployments, the RHEL Server for Physical or Virtual (RH00003) offering supports either one physical -or- two virtual deployments. Note that the RHEL Server for Physical and Virtual **does not** carry the “*virt-limit*” attribute, it is govern, by the “*instance multiplier*” attribute discussed earlier.

These new 2013 offerings significantly simplified deployment in clusters and migration. In high density clusters, the RHEL for Virtual Datacenters afforded the guests an easy migration strategy since the guests could migrate anywhere a certified hypervisor was covered by a vDC subscription. For low density clusters, the RHEL Server for Physical or Virtual subscription afforded the virtual guest deployments to migrate to any certified hypervisor platform since they are not dependent on the host for an entitlement.

Host/Guest Mapping and Virt-Who

When a subscription offers “bonus guest” entitlements there is a contractual obligation “for those guests to execute on the host from which they received their guest entitlement [1]”. So although the base subscription is “flexible” to be used on various certified hypervisor platforms, the guests are “not flexible” to be run anywhere - they must execute on the host from which their entitlement is offered.

Thus as a guest migrates, it must release the entitlement offered from its current host and attach the entitlement offered by its new host. This business rule caused some confusion since the older RHN tools, such as RHC Classic Hosted and Satellite 5, do not enforce this deployment expectation and thus allowed the guests to execute anywhere and access updates which were not in accordance with the contractual obligations of use.

In 2013 the “*host-limited*” subscription attribute was introduced and the newer subscription tools, such as Customer Portal, Satellite 6, and Subscription Asset Manager (SAM), provided by the RHSM suite, were updated to become sensitive to the “*host-limited*” attribute.

The “*host-limited*” attribute instructs the subscription tools to only offer the bonus guest entitlements to the virtual guests from the host where the guest is executing from (per the business rules of above [1]). In addition, the “*host-limited*” attribute guarantees that the physical entitlement be attached to the hypervisor before the bonus guest entitlements are available for offering to the guests.

For the subscription tools to assist the customer in fulfilling the business rule obligations [1], there needs to be a mapping between the hypervisors and the RHEL guests executing on those hypervisors. Since the guests are VMs and have no knowledge they are executing in a virtualized environment on a specific host, an external tool, virt-who, is required to create the mapping for the subscription service.

Further details on RHSM are available here: <https://access.redhat.com/articles/143253>

Further details on Virt-Who are available here: <https://access.redhat.com/articles/1300283>

Conclusion

Subscriptions entitle customers to Red Hat services through the Customer Portal and entitles end-systems to content access. The number of end-systems a subscription supports and the consumption rates may differ and thus the customer must be aware when viewing consumption reports in the subscription tools or creating SMA from on-premise use to ensure enough entitlements are available on-premise.

Red Hat offers many different subscriptions with different behaviors controlled through various attributes. The differences are not meant to confuse, but to offer unique value to meet changing customer needs and changes in Red Hat's portfolio.